(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 103577339 B
(45)授权公告日 2018.01.30

(21)申请号 201210264003.7

(22)申请日 2012.07.27

(65)同一申请的已公布的文献号
申请公布号 CN 103577339 A

(43)申请公布日 2014.02.12

(73)专利权人 深圳市腾讯计算机系统有限公司
地址 518057 广东省深圳市南山区高新区高新南一路飞亚达大厦5-10楼

(72)发明人 朱建平 李雅卿 许剑峰 李晨城 傅飞玲 朱柳嵩

(74)专利代理机构 广州三环专利商标代理有限公司 44202
代理人 郝传鑫

(51)Int.Cl.
G06F 12/06(2006.01)
G06F 17/30(2006.01)

(56)对比文件
CN 1936864 A,2007.03.28,
CN 101178693 A,2008.05.14,
US 6845427 B1,2005.01.18,
CN 101169761 A,2008.04.30,
US 2011/0283044 A1,2011.11.17,

审查员 唐进岭

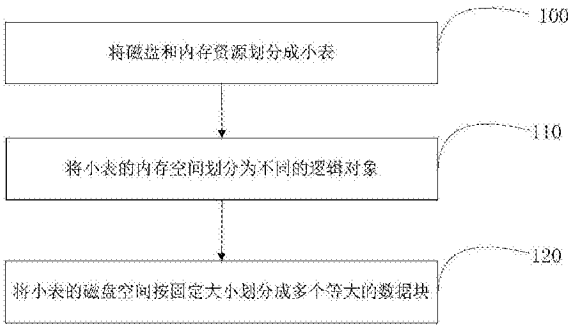
权利要求书2页 说明书7页 附图4页

(54)发明名称

一种数据存储方法及系统

(57)摘要

本发明属于计算机技术领域，尤其涉及一种数据存储方法及系统。本发明实施例的数据存储方法包括：将磁盘和内存资源划分成小表；将小表的内存空间划分为不同的逻辑对象；将小表的磁盘空间按固定大小划分成多个等大的数据块。本发明实施例的数据存储系统及方法通过划分存储服务器上的磁盘和内存资源为一个个独立的小表，将小表用作业务资源分配和管理的基本单元，可实现单机资源在多个业务上的复用；另外，混合式索引及其关联的合并写，块回收技术在提升系统随机写IOPS的同时能大幅节省了索引的内存空间。



CN 103577339 B

1.一种数据存储方法，包括：

将磁盘和内存资源划分成小表；

将小表的内存空间划分为不同的逻辑对象，所述小表内存空间划分的逻辑对象包括：记录索引缓存、写缓存和块缓存；所述记录索引缓存用于存储记录的索引信息；所述写缓存用于记录写缓冲；所述块缓存由块结构描述符组成，用于统计块状态信息；

将小表的磁盘空间按固定大小划分成多个等大的数据块；

所述数据存储方法的块合并写包括：定时从写缓存中取记录；对每个记录读取关键词缓存及磁盘，判断记录写场景；统计结果集，判断写入数据能否凑成一个数据块，如果写入数据能够凑成一个块，将结果集中记录整理成一个块；如果写入数据不能凑成一个块，则继续处理剩下的记录。

2.根据权利要求1所述的数据存储方法，其特征在于，所述记录索引缓存包括桶索引和大记录索引，大于设定的记录大小阀值的记录采用大记录索引，在大记录索引中，记录采用独立索引；小于设定的记录大小阀值的记录采用桶索引，将桶打包成记录存储，形成桶记录；所述大于设定的记录大小阀值的记录为大记录，所述小于设定的记录大小阀值的记录为小记录。

3.根据权利要求1所述的数据存储方法，其特征在于，所述数据块的数据组织结构包括序列和定长信息块，所述序列用于描述存储在数据块内的记录，所述定长信息块用于描述块的元信息，所述元信息包括块的数据校验、块写入时间和块内的记录数。

4.根据权利要求2所述的数据存储方法，其特征在于，所述数据存储方法的记录读包括：读写缓存，根据关键字从写缓存查找记录，如果找到记录，直接返回；从记录索引缓存获取记录索引，根据所述记录索引获取记录偏移地址和记录大小；根据索引中的偏移地址和记录大小，从磁盘读取数据。

5.根据权利要求2所述的数据存储方法，其特征在于，所述数据存储方法的记录修改包括：读写缓存，根据关键词从写缓存中查找记录；判断是否找到相应记录，如果找到记录，在写缓存中更新记录并返回；如果没有找到记录，则将记录添加到写缓存。

6.根据权利要求2所述的数据存储方法，其特征在于，所述数据存储方法的块合并写判断记录写场景包括：如果是小记录更新或删除，更新后记录小于一预定值，则根据桶记录的尺寸将记录转成桶记录后添加到结果集或将桶记录重新序列化后添加到结果集；如果是小记录更新，更新后此记录大于一预定值，将记录从桶记录中删除，将桶记录重新序列化后添加到结果集，并构造二个索引更新对象，添加到索引更新集合；如果是大记录更新，将记录添加到结果集，构造索引更新对象，添加到索引更新集合；如果是大记录删除，构造删除流水，添加到结果集，构造索引更新对象，添加到索引更新集合。

7.根据权利要求3所述的数据存储方法，其特征在于，所述数据存储方法的块合并写包括：根据块缓存找到2个有效数据长度均小于预定大小的数据块；从磁盘上读取找到的数据块；解析数据块，根据关键词缓存信息剔除过时数据，将有效数据合并成一个数据块；重新计算新数据块的定长信息块，写入时间设置为两个老数据块中较新的时间戳；写入新的数据块；在关键词缓存中更新新数据块中记录的索引；在块缓存中重置两个老的数据块信息，更新新写入的数据块的信息。

8.一种数据存储系统，包括磁盘和内存资源，其特征在于，还包括资源划分模块、逻辑

对象划分模块和数据块划分模块，所述资源划分模块将磁盘和内存资源划分成小表；逻辑对象划分模块将小表的内存空间划分为不同的逻辑对象，所述小表内存空间划分的逻辑对象包括：记录索引缓存、写缓存和块缓存；所述记录索引缓存用于存储记录的索引信息；所述写缓存用于记录写缓冲；所述块缓存由块结构描述符组成，用于统计块状态信息；数据块划分模块将小表的磁盘空间按固定大小划分成多个等大的数据块；

所述数据存储系统还包括块合并写模块，所述块合并写模块用于定时从写缓存中取记录；对每个记录读取关键词缓存及磁盘，判断记录写场景；统计结果集，判断写入数据能否凑成一个数据块，如果写入数据能够凑成一个块，将结果集中记录整理成一个块；如果写入数据不能凑成一个块，则继续处理剩下的记录。

9.根据权利要求8所述的数据存储系统，其特征在于，所述记录索引缓存包括桶索引和大记录索引，大于设定的记录大小阀值的记录采用大记录索引，在大记录索引中，记录采用独立索引；小于设定的记录大小阀值的记录采用桶索引，将桶打包成记录存储。

10.根据权利要求8或9所述的数据存储系统，其特征在于，所述数据块的数据组织结构包括序列和定长信息块，所述序列用于描述存储在数据块内的记录，所述定长信息块用于描述块的元信息，所述元信息包括块的数据校验、块写入时间和块内的记录数。

11.一种数据存储服务器，包括磁盘和内存资源，所述磁盘和内存资源划分成小表；所述小表的内存空间划分为不同的逻辑对象；所述小表内存空间划分的逻辑对象包括：记录索引缓存、写缓存和块缓存；所述记录索引缓存用于存储记录的索引信息；所述写缓存用于记录写缓冲；所述块缓存由块结构描述符组成，用于统计块状态信息；小表的磁盘空间按固定大小划分成多个等大的数据块；

所述数据存储服务器进行块合并写包括：定时从写缓存中取记录；对每个记录读取关键词缓存及磁盘，判断记录写场景；统计结果集，判断写入数据能否凑成一个数据块，如果写入数据能够凑成一个块，将结果集中记录整理成一个块；如果写入数据不能凑成一个块，则继续处理剩下的记录。

12.根据权利要求11所述的数据存储服务器，其特征在于，所述数据块的数据组织结构包括序列和定长信息块，所述序列用于描述存储在数据块内的记录，所述定长信息块用于描述块的元信息。

一种数据存储方法及系统

技术领域

[0001] 本发明属于计算机技术领域，尤其涉及一种数据存储方法及系统。

背景技术

[0002] 当前分布式数据存储系统的存储介质主要有SATA盘、SAS盘、SSD盘/卡，随着硬件制造技术的发展，单盘的存储容量在不断提升，但盘的随机IO(输入输出)能力没能成比例的提升，存储介质的随机IO能力是一个潜在的性能瓶颈。解决存储介质的随机IO性能瓶颈的思路主要是将随机IO转为顺序IO或引入缓存减少IO次数。由于在大部分业务存储访问场景下随机读难以避免，一般通过引入内存Cache减少对存储介质的访问或变更存储介质，如改用随机读性能较高的SSD盘/卡加以优化。随机写IO的优化可将随机写变为顺序写(如BigTable的SSTable)或mmap将数据映射到内存进行异步IO。

[0003] 现有其他提升随机IO的方式有如下几种:Google BigTable通过随机写转顺序写实现IO性能的优化，通过MemTable和SSTable的方式，将一段时间的所有数据更新存储在一起顺序写入磁盘，之后再将记录按照预定义的顺序进行分拆并跟老版本的数据进行合并；TyotoCabinet采用mmap将磁盘上的数据映射到共享内存中，通过内存来减少对磁盘的读写；MySQL的InnoDB存储引擎在底层采用B+树的组织方式，同样通过Buffer Pool(缓冲池)将写转化为异步写，来提升写延时体验和减少对磁盘的读写。

[0004] 现有提升随机IO的方式缺点在于:随机写转顺序写系统实现复杂，运营成本高；采取MemTable/SSTable将随机写全部转成顺序写的方式，在数据读取时可能需要读取多处以获得最新的数据版本；此外，在做数据合并时需执行较大规模的数据读写，并可能伴随分裂等操作，系统比较复杂，运营维护成本较高；针对mmap和MySQL的buffer pool方案，如果单机的存储量高于内存，数据无明显热点时，系统的整体IO性能转为存储介质的实际能力，内存对IO性能提升效率比较有限。

发明内容

[0005] 本发明提供了一种数据存储方法及系统，旨在解决现有技术数据存储方式随机输入输出能力有限、实现复杂以及运营困难的问题。

[0006] 本发明是这样实现的，一种数据存储方法，包括：

[0007] 将磁盘和内存资源划分成小表；

[0008] 将小表的内存空间划分为不同的逻辑对象；

[0009] 将小表的磁盘空间按固定大小划分成多个等大的数据块。

[0010] 本发明实施例的技术方案还包括：所述小表内存空间划分的逻辑对象包括：记录索引缓存、写缓存和块缓存；所述记录索引缓存用于存储记录的索引信息；所述写缓存用于记录写缓冲；所述块缓存由块结构描述符组成，用于统计块状态信息。

[0011] 本发明实施例的技术方案还包括：所述记录索引缓存包括桶索引和大记录索引，大于设定的记录大小阀值的记录采用大记录索引，在大记录索引中，记录采用独立索引；小

于设定的记录大小阀值的记录采用桶索引，将桶打包成记录存储。

[0012] 本发明实施例的技术方案还包括：所述数据块的数据组织结构包括序列和定长信息块，所述序列用于描述存储在数据块内的记录，所述定长信息块用于描述描述块的元信息。

[0013] 本发明实施例的技术方案还包括：所述数据存储方法的记录读包括：读写缓存，根据关键字从写缓存查找记录，如果找到记录，直接返回；从记录索引缓存获取记录索引，根据所述记录索引获取记录偏移地址和记录大小；根据索引中的偏移地址和记录大小，从磁盘读取数据。

[0014] 本发明实施例的技术方案还包括：所述数据存储方法的记录修改包括：读写缓存，根据关键词从写缓存中查找记录；判断是否找到相应记录，如果找到记录，在写缓存中更新记录并返回；如果没有找到记录，则将记录添加到写缓存。

[0015] 本发明实施例的技术方案还包括：所述数据存储方法的块合并写包括：定时从写缓存中取记录；对每个记录读取关键词缓存及磁盘，判断记录写场景；统计结果集，判断写入数据能否凑成一个数据块，如果写入数据可以凑成一个块，将结果集中记录整理成一个块；如果写入数据不能凑成一个块，则继续处理剩下的记录。

[0016] 本发明实施例的技术方案还包括：所述数据存储方法的块合并写判断记录写场景包括：如果是小记录更新或删除，更新后记录小于一预定值，则根据桶记录的尺寸将记录转成桶记录后添加到结果集或将桶记录重新序列化后添加到结果集；如果是小记录更新，更新后此记录大于一预定值，将记录从桶记录中删除，将桶记录重新序列化后添加到结果集，并构造二个索引更新对象，添加到索引更新集合；如果是大记录更新，将记录添加到结果集，构造索引更新对象，添加到索引更新集合；如果是大记录删除，构造删除流水，添加到结果集，构造索引更新对象，添加到索引更新集合。

[0017] 本发明实施例的技术方案还包括：所述数据存储方法的块合并写包括：根据块缓存找到2个有效数据长度均小于预定大小的数据块；从磁盘上读取找到的数据块；解析数据块，根据关键词缓存信息剔除过时数据，将有效数据合并成一个数据块；重新计算新数据块的定长信息块，写入时间设置为两个老数据块中较新的时间戳；写入新的数据块；在关键词缓存中更新新数据块中记录的索引；在块缓存中重置两个老的数据块信息，更新新写入的数据块的信息。

[0018] 本发明实施例采取的另一技术方案为：一种数据存储系统，包括磁盘、内存、资源划分模块、逻辑对象划分模块和数据块划分模块，所述资源划分模块将磁盘和内存资源划分成小表；逻辑对象划分模块将小表的内存空间划分为不同的逻辑对象；数据块划分模块小表的磁盘空间按固定大小划分成多个等大的数据块。

[0019] 本发明实施例的技术方案还包括：所述小表内存空间划分的逻辑对象包括：记录索引缓存、写缓存和块缓存；所述记录索引缓存用于存储记录的索引信息；所述写缓存用于记录写缓冲；所述块缓存由块结构描述符组成，用于统计块状态信息。

[0020] 本发明实施例的技术方案还包括：所述记录索引缓存包括桶索引和大记录索引，大于设定的记录大小阀值的记录采用大记录索引，在大记录索引中，记录采用独立索引；小于设定的记录大小阀值的记录采用桶索引，将桶打包成记录存储。

[0021] 本发明实施例的技术方案还包括：所述数据块的数据组织结构包括序列和定长信

息块，所述序列用于描述存储在数据块内的记录，所述定长信息块用于描述描述块的元信息。

[0022] 本发明实施例采取的另一技术方案为：一种数据存储服务器，包括磁盘和内存，所述磁盘和内存资源划分成小表；所述小表的内存空间划分为不同的逻辑对象；所述磁盘空间按固定大小划分成多个等大的数据块。

[0023] 本发明实施例的技术方案还包括：所述小表内存空间划分的逻辑对象包括：记录索引缓存、写缓存和块缓存；所述记录索引缓存用于存储记录的索引信息；所述写缓存用于记录写缓冲；所述块缓存由块结构描述符组成，用于统计块状态信息。

[0024] 本发明实施例的技术方案还包括：所述数据块的数据组织结构包括序列和定长信息块，所述序列用于描述存储在数据块内的记录，所述定长信息块用于描述描述块的元信息。

[0025] 本发明实施例的技术方案具有如下优点或有益效果：本发明实施例的数据存储系统及方法通过划分存储服务器上的磁盘和内存资源为一个个独立的小表，将小表用作业务资源分配和管理的基本单元，可实现单机资源在多个业务上的复用；另外，混合式索引及其关联的合并写，块回收技术在提升系统随机写IOPS的同时能大幅节省了索引的内存空间。

附图说明

[0026] 附图1是本发明实施例的数据存储方法的流程图；

[0027] 附图2是本发明实施例的数据存储方法的记录索引缓存的内存结构示意图；

[0028] 附图3是本发明实施例的数据存储方法的块缓存的内存结构示意图；

[0029] 附图4是本发明实施例的数据存储方法的记录读的流程图；

[0030] 附图5是本发明实施例的数据存储方法的记录修改的流程图；

[0031] 附图6是本发明实施例的数据存储方法的块合并写的流程图；

[0032] 附图7是本发明实施例的数据存储方法的块回收的流程图；

[0033] 附图8是本发明实施例的数据存储系统的结构示意图；

[0034] 附图9是本发明实施例的数据存储服务器的结构示意图。

具体实施方式

[0035] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[0036] 请参阅图1，是本发明实施例的数据存储方法的流程图。本发明实施例的数据存储方法包括以下步骤：

[0037] 步骤100：将磁盘和内存资源划分成小表(Tablet)；

[0038] 在步骤100中，每个小表独立工作，小表通过Hash哈希和记录索引相结合的方式对磁盘中的记录进行索引。在本发明的实施方式中，假设一个小表的资源是由磁盘上物理地址连续的2GB空间和内存中16MB的共享内存空间组成，另外，可以根据需要设定磁盘空间和内存空间的大小。

[0039] 步骤110：将小表的内存空间划分为不同的逻辑对象；

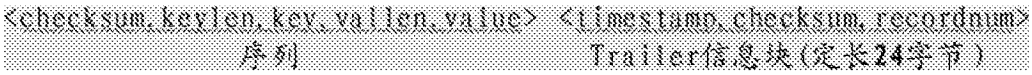
[0040] 在步骤110中，小表的内存空间划分为3个逻辑对象：记录索引缓存KeyCache、写缓存WriteCache和块缓存BlockCache，其中：KeyCache占用12MB空间，包括桶索引和大记录索引两部分，具体请参阅图2，是本发明实施例的数据存储方法的记录索引缓存的内存结构示意图，其用于存储记录的索引信息，记录对应的也分为桶记录和大记录两种类型，在本发明另一实施方式中，KeyCache可以仅包括大记录索引部分，记录也可以仅为大记录一种类型。记录的索引查找需要计算hash(key)%(300*1024)的值，从而获取到一条桶记录索引，根据桶记录索引的桶头字段定位到一个大记录索引，用大记录索引的next字段遍历整条链，比对key查找到记录的索引；若大记录索引没有命中，取桶记录索引为记录的索引。由于Keycache的组织采用大记录索引和桶索引的混合索引模式，一方面克服了全索引模式的缺点，全索引需要为每条记录在内存中建条索引，其索引存储空间大，对内存需求大；另一方面，克服了全桶索引的缺点，桶记录是采用hash结构，全桶索引更新桶中的一条子记录会将整个桶记录读出并写入，存在磁盘读写带宽的放大，对SSD存储介质来说，写带宽放大会降低寿命，同时对读写性能也有所影响。本发明实施例的数据存储方法根据内存的情况，制定一个记录大小阀值，大于此阀值大小的记录采用大记录索引，在大记录索引中一条记录一个独立索引，用链式串联起来；小于此阀值的记录采用桶记录，将Hash的桶打包成一个记录存储。在磁盘中，一个Hash桶在内存中建立一条索引，极大降低了索引的数量。在本发明实施例中，可以将小表设计成2G，桶记录和大记录索引各约30万个，大于4KB的记录用大记录索引，大于4KB的记录采用桶索引。如果记录大小都比较大，2G的小表存储的记录数较少，30万索引已足够存储；如记录都比较小，30万的桶索引可以保证每个桶记录的大小不超过4KB。

[0041] WriteCache占用约4MB空间，用做记录写缓冲，实现数据异步写入磁盘，BlockCache占用64KB，由4000个16字节的块结构描述符组成，用作统计块状态信息，具体请参阅图3，是本发明实施例的数据存储方法的块缓存的内存结构示意图。块的有效size字段表示，块中记录未发生更新的记录的总大小；块更新时间字段表示块写入磁盘的时间；块next字段用于将空闲块和待回收块用链表组织起来，链接成空闲块链(free链)和回收块链(recycle链)。WriteCache是基于共享内存的哈希图(HashMap)。

[0042] 步骤120：将小表的磁盘空间按固定大小划分成多个等大的数据块。

[0043] 在步骤120中，将小表2GB的磁盘空间按固定大小划分成多个等大的数据块，在本发明实施方式中，假设块大小为512KB，每个数据块的数据组织结构如下：

[0044]



[0045] 每个<checksum,keylen,key,vallen,value>5元组描述存储在数据块内的一条记录，记录在数据块内紧凑排列，在块的尾部用定长信息块(Trailer信息块)描述块的元信息，如块的数据校验、块写入时间和块内的记录数。

[0046] 在本发明实施方式中，数据存储方法的典型数据读写流程包括记录读(read)、记录修改(包括Insert/Update/Delete)、块合并写和块回收；具体请一并参阅图4，是本发明实施例的数据存储方法的记录读的流程图。本发明数据存储方法的记录读包括以下步骤：

[0047] 步骤200：读WriteCache，根据key关键字从WriteCache查找记录，如果找到记录，直接返回；

[0048] 步骤210：从KeyCache获取记录索引，根据上述记录索引查找方法获取到记录偏移地址(offset)和大小；

[0049] 步骤220：根据索引中的偏移地址和记录大小，从磁盘读取数据；

[0050] 在步骤220中，根据记录的类型做相应处理：如果记录是大记录，直接返回；如果记录是桶记录，则解析桶记录，遍历桶记录中的所有子记录，匹配key，找到key返回记录；如果没有找到，则返回记录不存在的错误提示。

[0051] 请一并参阅图5，是本发明实施例的数据存储方法的记录修改的流程图。本发明数据存储方法的记录修改包括以下步骤：

[0052] 步骤300：读WriteCache，根据key从WriteCache查找记录；

[0053] 步骤310：判断是否找到相应记录，如果找到记录，则进入步骤320；如果没有找到记录则进入步骤330；

[0054] 步骤320：在WriteCache中更新记录并返回；

[0055] 步骤330：将记录添加到WriteCache。

[0056] 在步骤330中，删除记录可通过设置记录标志加以区分。

[0057] 请一并参阅图6，是本发明实施例的数据存储方法的块合并写的流程图。本发明数据存储方法的块合并写包括以下步骤：

[0058] 步骤400：定时从write cache按先进先出(FIFO)顺序取记录；

[0059] 在步骤400中，取记录一次取512KB数据。

[0060] 步骤410：对每个记录，读取key cache及ssd盘，判断记录写场景：

[0061] 如果是小记录更新，更新后此记录仍小于4kb：转步骤420

[0062] 如果是小记录更新，更新后此记录大于4kb：转步骤430

[0063] 如果是大记录更新：转步骤440

[0064] 如果是小记录删除：转步骤420

[0065] 如果是大记录删除：转步骤450

[0066] 步骤420：根据桶记录的size作如下判断：

[0067] size==0：将此记录转成桶记录后添加到结果集；

[0068] size>0：根据桶记录的<offset,size>从ssd盘读取记录，反序列化，查找并更新此记录，将桶记录重新序列化后添加到结果集，构造一个索引更新对象，添加到索引更新集合，转步骤460。

[0069] 步骤430：将记录添加到结果集。

[0070] 将记录从桶记录中删除，将桶记录重新序列化后添加到结果集，并构造二个索引更新对象，添加到索引更新集合，转步骤460。

[0071] 步骤440：将记录添加到结果集，构造一个索引更新对象，添加到索引更新集合，转步骤460。

[0072] 步骤450：构造一个删除流水，添加到结果集；构造一个索引更新对象，添加到索引更新集合，转步骤460。

[0073] 步骤460：统计结果集，判断写入数据能否凑成一个数据块(512KB)，如果能凑成一个块，转步骤470；如果不能凑成一个块，转步骤410，继续处理剩下的记录；

[0074] 步骤470：将结果集中记录整理成一个块，计算Trailer信息块，整块写入ssd；将索

引更新集合提交，批量更新key cache；提交清理write cache，将已写入记录从write cache中清除。

[0075] 在步骤470中，清理结果集和索引更新集合，返回步骤410继续处理剩下的记录。

[0076] 请一并参阅图7，是本发明实施例的数据存储方法的块回收的流程图。本发明数据存储方法的块回收包括以下步骤：

[0077] 步骤500：根据block cache找到2个有效数据长度均小于256KB的数据块；

[0078] 步骤510：从ssd盘上读取步骤500中选取的2个数据块；

[0079] 步骤520：解析2个数据块，根据key cache信息剔除过时数据，将有效数据合并成一个数据块；

[0080] 步骤530：重新计算新数据块的Trailer信息块，写入时间设置为两个老数据块中较新的时间戳；

[0081] 步骤540：写入新的数据块；

[0082] 步骤550：在key cache中更新新数据块中记录的索引；在block cache中重置两个老的数据块信息，更新新写入的数据块的信息。

[0083] 请参阅图8，是本发明实施例的数据存储系统的结构示意图。本发明实施例的数据存储系统包括磁盘、内存、资源划分模块、逻辑对象划分模块和数据块划分模块。

[0084] 资源划分模块将磁盘和内存资源划分成小表(Tablet)，其中，每个小表独立工作，小表通过Hash哈希和记录索引相结合的方式对磁盘中的记录进行索引。在本发明的实施方式种，假设一个小表的资源是由磁盘上物理地址连续的2GB空间和内存中16MB的共享内存空间组成。

[0085] 逻辑对象划分模块将小表的内存空间划分为不同的逻辑对象，表的内存空间划分为3个逻辑对象：记录索引缓存KeyCache、写缓存WriteCache和块缓存BlockCache，其中：KeyCache占用12MB空间，包括桶索引和大记录索引两部分，其用于存储记录的索引信息，记录对应的也分为桶记录和大记录两种类型，在本发明另一实施方式中，KeyCache可以仅包括大记录索引部分，记录也可以仅为大记录一种类型。记录的索引查找需要计算hash(key)%(300*1024)的值，从而获取到一条桶记录索引，根据桶记录索引的桶头字段定位到一个大记录索引，用大记录索引的next字段遍历整条链，比对key查找到记录的索引；若大记录索引不命中，取桶记录索引为记录的索引。WriteCache占用约4MB空间，用做记录写缓冲，实现数据异步落磁盘，BlockCache占用64KB，由4000个16字节的块结构描述符组成，用作统计块状态信息。块的有效size字段表示，块中记录未发生更新的记录的总大小；块更新时间字段表示块写入磁盘的时间；块next字段用于将空闲块和待回收块用链表组织起来，链接成空闲块链(free链)和回收块链(recycle链)。WriteCache是基于共享内存的哈希图(HashMap)。

[0086] 数据块划分模块将小表的磁盘空间按固定大小划分成多个等大的数据块，在本发明实施方式中，假设块大小为512KB，每个数据块的数据组织结构如下：

[0087]

<checksum,keylen,key,vallen,value> <timestamp,checksum,recordnum>
序列 Trailer信息块(定长24字节)

[0088] 每个<checksum,keylen,key,vallen,value>5元组描述存储在数据块内的一条记

录，记录在数据块内紧凑排列，在块的尾部用定长信息块(Trailer信息块)描述块的元信息，如块的数据校验、块写入时间和块内的记录数。

[0089] 请参阅图9，是本发明实施例的数据存储服务器的结构示意图。本发明实施例的数据存储系统包括磁盘和内存。

[0090] 磁盘和内存资源划分成小表(Tablet)，其中，每个小表独立工作，小表通过Hash哈希和记录索引相结合的方式对磁盘中的记录进行索引。在本发明的实施方式种，假设一个小表的资源是由磁盘上物理地址连续的2GB空间和内存中16MB的共享内存空间组成。

[0091] 小表的内存空间划分为不同的逻辑对象，表的内存空间划分为3个逻辑对象：记录索引缓存KeyCache、写缓存WriteCache和块缓存BlockCache，其中：KeyCache占用12MB空间，包括桶索引和大记录索引两部分，其用于存储记录的索引信息，记录对应的也分为桶记录和大记录两种类型，在本发明另一实施方式中，KeyCache可以仅包括大记录索引部分，记录也可以仅为大记录一种类型。记录的索引查找需要计算hash(key)%(300*1024)的值，从而获取到一条桶记录索引，根据桶记录索引的桶头字段定位到一个大记录索引，用大记录索引的next字段遍历整条链，比对key查找到记录的索引；若大记录索引不命中，取桶记录索引为记录的索引。WriteCache占用约4MB空间，用做记录写缓冲，实现数据异步落磁盘，BlockCache占用64KB，由4000个16字节的块结构描述符组成，用作统计块状态信息。块的有效size字段表示，块中记录未发生更新的记录的总大小；块更新时间字段表示块写入磁盘的时间；块next字段用于将空闲块和待回收块用链表组织起来，链接成空闲块链(free链)和回收块链(recycle链)。WriteCache是基于共享内存的哈希图(HashMap)。

[0092] 小表的磁盘空间按固定大小划分成多个等大的数据块，在本发明实施方式中，假设块大小为512KB，每个数据块的数据组织结构如下：

[0093]

[0094] 每个<checksum,keylen,key,vallen,value>5元组描述存储在数据块内的一条记录，记录在数据块内紧凑排列，在块的尾部用定长信息块(Trailer信息块)描述块的元信息，如块的数据校验、块写入时间和块内的记录数。

[0095] 本发明实施例的数据存储系统及方法通过划分存储服务器上的磁盘和内存资源为一个个独立的小表，将小表用作业务资源分配和管理的基本单元，可实现单机资源在多个业务上的复用；另外，本发明实施例的数据存储系统及方法通过合并写、块回收的方式减少对磁盘IO的随机写入；读取记录时，根据内存中的记录索引通过一次磁盘IO实现记录的获取。

[0096] 以上所述仅为本发明的较佳实施例而已，并不用以限制本发明，凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等，均应包含在本发明的保护范围之内。

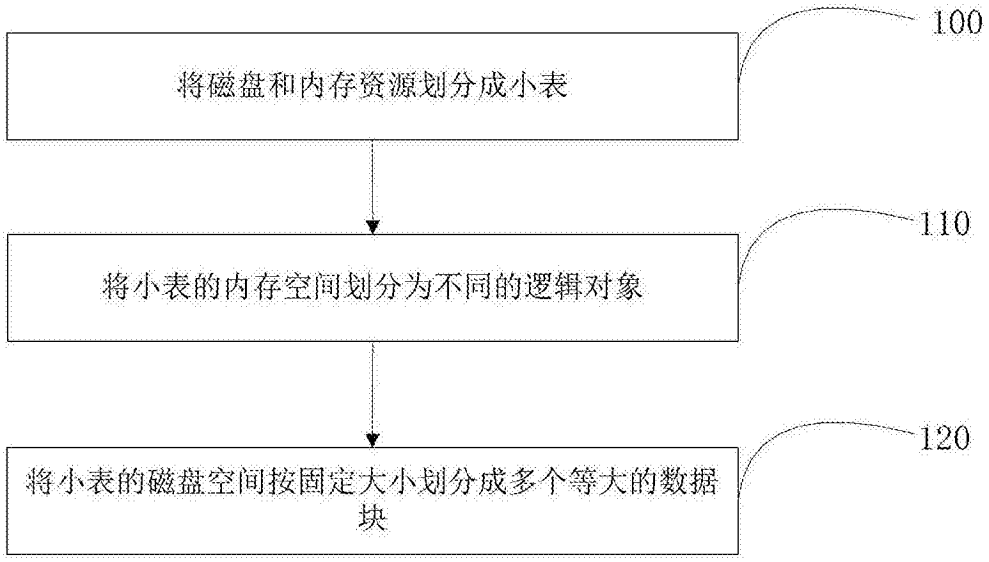

100
将磁盘和内存资源划分成小表
110
将小表的内存空间划分为不同的逻辑对象
120
将小表的磁盘空间按固定大小划分成多个等大的数据块

图1

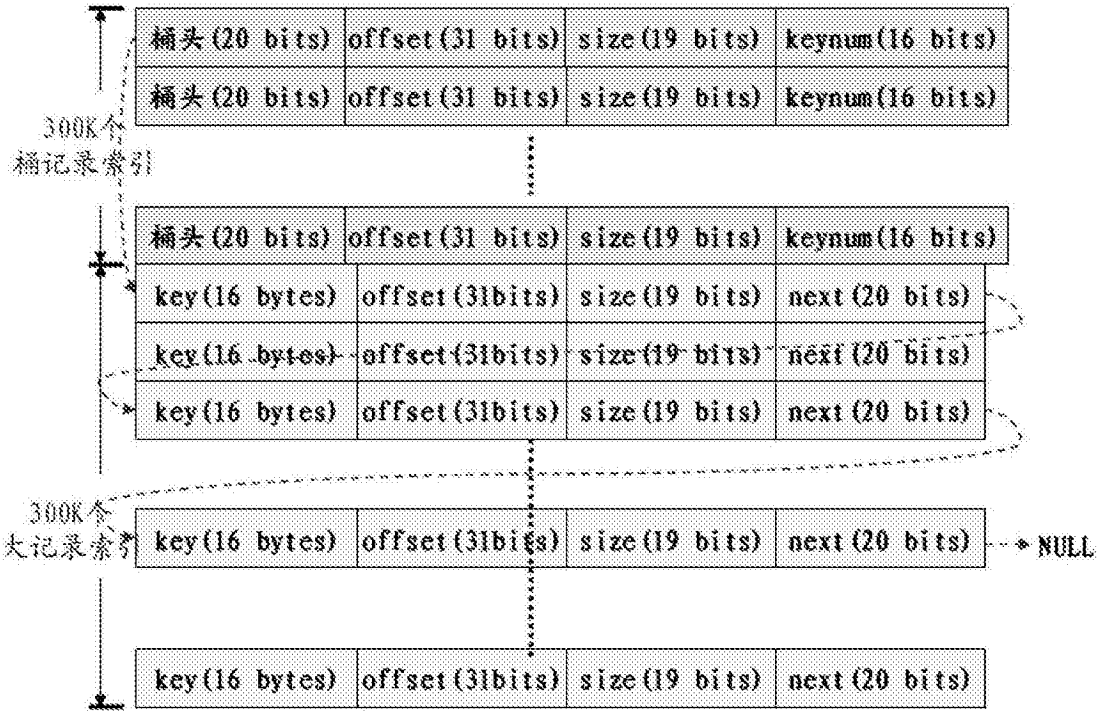

桶头(20 bits)
offset(31 bits)
size(19 bits)
keynum(16 bits)
300K个桶记录索引
key(16 bytes)
offset(31bits)
size(19 bits)
next(20 bits)
300K个大记录索引
NULL

图2

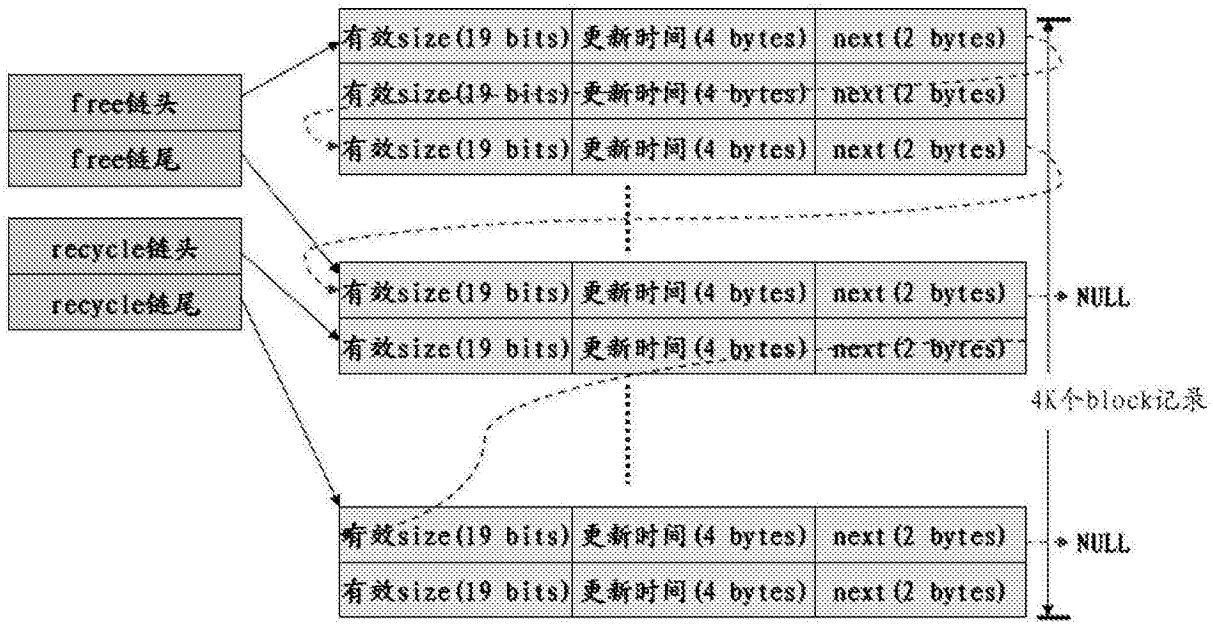

free链头
free链尾
recycle链头
recycle链尾
有效size(19 bits)
更新时间(4 bytes)
next(2 bytes)
NULL
4K个block记录

图3

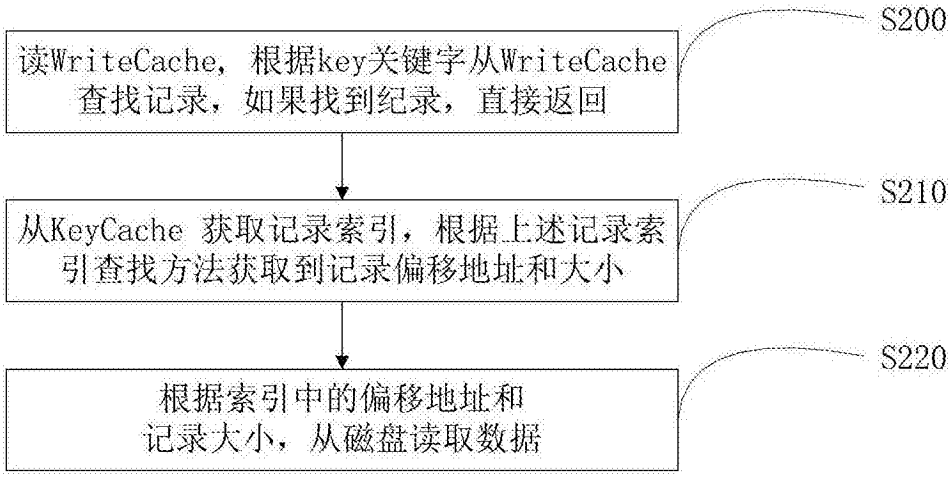

读WriteCache，根据key关键字从WriteCache查找记录，如果找到纪录，直接返回
S200
从KeyCache 获取记录索引，根据上述记录索引查找方法获取到记录偏移地址和大小
S210
根据索引中的偏移地址和记录大小，从磁盘读取数据
S220

图4

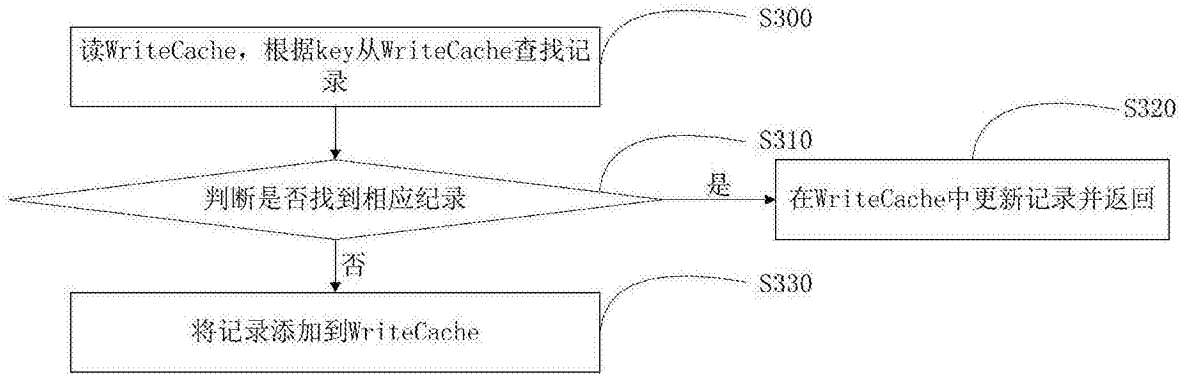

S300
读WriteCache，根据key从WriteCache查找记录
S310
判断是否找到相应纪录
是
S320
在WriteCache中更新记录并返回
否
S330
将记录添加到WriteCache

图5

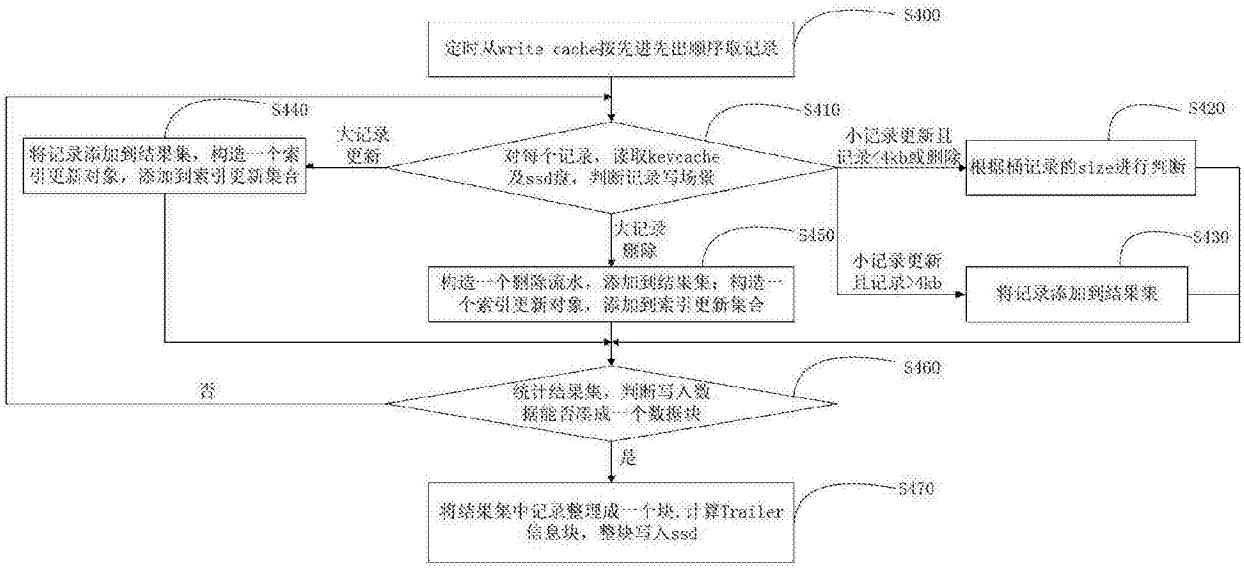

S400
定时从write cache按先进先出顺序取记录
S410
对每个记录，读取keycache及ssd盘，判断记录写场景
大记录更新
小记录更新且记录<4kb或删除
大记录删除
小记录更新且记录>4kb
S420
根据桶记录的size进行判断
S430
将记录添加到结果集
S440
将记录添加到结果集，构造一个索引更新对象，添加到索引更新集合
S450
构造一个删除流水，添加到结果集；构造一个索引更新对象，添加到索引更新集合
S460
统计结果集，判断写入数据能否凑成一个数据块
否
是
S470
将结果集中记录整理成一个块，计算Trailer信息块，整块写入ssd

图6

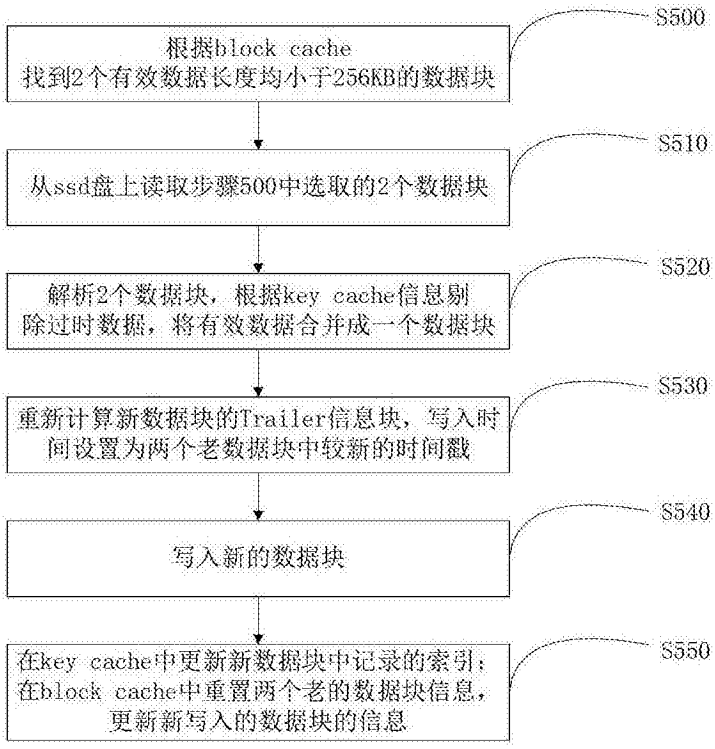

根据block cache
找到2个有效数据长度均小于256KB的数据块
S500
从ssd盘上读取步骤500中选取的2个数据块
S510
解析2个数据块，根据key cache信息剔除过时数据，将有效数据合并成一个数据块
S520
重新计算新数据块的Trailer信息块，写入时间设置为两个老数据块中较新的时间戳
S530
写入新的数据块
S540
在key cache中更新新数据块中记录的索引；在block cache中重置两个老的数据块信息，更新新写入的数据块的信息
S550

图7

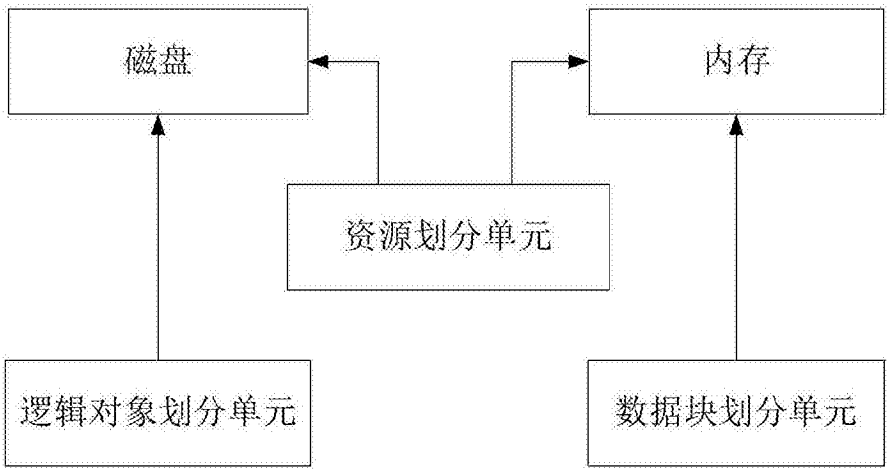

磁盘
内存
资源划分单元
逻辑对象划分单元
数据块划分单元

图8

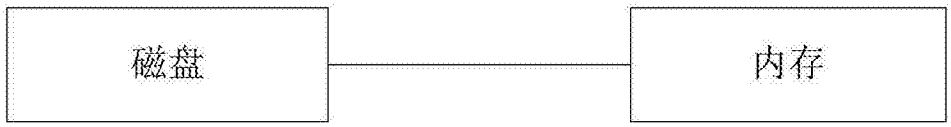

磁盘
内存

图9